

CLAIMS

WHAT IS CLAIMED IS:

- 1 1. A method for supporting a platform independent object format for a run-time
2 environment, comprising the computer-implemented steps of:
3 accessing a definition of an object in terms of a composition of one or more primitive
4 types;
5 accessing a platform-specific description of layout parameters of the one or more
6 primitive types; and
7 generating a layout for the object in a high-order language based on the definition of
8 the object and the platform-specific description.
- 1 2. The method according to claim 1, further comprising the step of generating
2 instructions for an accessor operation to access a slot in the object holding a value for one
3 of the one or more primitive types.
- 1 3. The method according to claim 1, further comprising the step of generating
2 instructions for a get operation to fetch a value for one of the one or more primitive types
3 from a slot in the object.
- 1 4. The method according to claim 1, further comprising the step of generating
2 instructions for a set operation to store a value for one of the one or more primitive types
3 from a slot in the object.
- 1 5. The method according to claim 1, wherein the one or more primitive types includes
2 or more of the following types: integer, floating point, and reference.

1 6. The method according to claim 5, wherein the primitive reference type is one of a
2 native machine pointer type and a numeric reference type.

1 7. The method according to claim 1, wherein the layout parameters include a size and
2 an alignment of the primitive types.

1 8. A method for supporting an object format for a plurality of incompatible platforms
2 for a run-time environment, comprising the computer-implemented steps of:
3 accessing a definition of an object as a plurality of slots containing a primitive type;
4 accessing a plurality of platform-specific descriptions of layout parameters of the one
5 or more primitive types, said platform-specific descriptions corresponding
6 respectively to the incompatible platforms; and
7 generating a plurality of layouts, corresponding respectively to the incompatible
8 platforms, for the object in a high-order language based on the definition of the
9 object and the platform-specific descriptions.

1 9. The method according to claim 8, where the slots are located in the layouts for the
2 incompatible platforms, when compiled by a corresponding platform-specific compiler,
3 at same offsets.

1 10. A computer-readable medium bearing instructions for supporting a platform
2 independent object format for a run-time environment, said instructions being arranged to
3 cause one or more processors upon execution thereby to perform the steps of:
4 accessing a definition of an object in terms of a composition of one or more primitive
5 types;

6 accessing a platform-specific description of layout parameters of the one or more
7 primitive types; and
8 generating a layout for the object in a high-order language based on the definition of
9 the object and the platform-specific description.

1 11. The computer-readable medium according to claim 10, wherein said instructions
2 are further arranged for performing the step of generating instructions for an accessor
3 operation to access a slot in the object holding a value for one of the one or more
4 primitive types.

1 12. The computer-readable medium according to claim 10, wherein said instructions
2 are further arranged for performing the step of generating instructions for a get operation
3 to fetch a value for one of the one or more primitive types from a slot in the object.

1 13. The computer-readable medium according to claim 10, wherein said instructions
2 are further arranged for performing the step of generating instructions for a set operation
3 to store a value for one of the one or more primitive types from a slot in the object.

1 14. The computer-readable medium according to claim 10, wherein the one or more
2 primitive types includes or more of the following types: integer, floating point, and
3 reference.

1 15. The computer-readable medium according to claim 14, wherein the primitive
2 reference type is one of a native machine pointer type and a numeric reference type.

1 16. The computer-readable medium according to claim 10, wherein the layout
2 parameters include a size and an alignment of the primitive types.

1 17. A computer-readable medium bearing instructions for supporting an object
2 format for a plurality of incompatible platforms for a run-time environment, said
3 instructions being arranged to cause one or more processors upon execution thereby to
4 perform the steps of:
5 accessing a definition of an object as a plurality of slots containing a primitive type;
6 accessing a plurality of platform-specific descriptions of layout parameters of the one
7 or more primitive types, said platform-specific descriptions corresponding
8 respectively to the incompatible platforms; and
9 generating a plurality of layouts, corresponding respectively to the incompatible
10 platforms, for the object in a high-order language based on the definition of the
11 object and the platform-specific descriptions.

1 18. The computer-readable medium according to claim 17, wherein the slots are
2 located in the layouts for the incompatible platforms, when compiled by a corresponding
3 platform-specific compiler, at same offsets.

*add
B'*